

## Hand-in assignments (Thursday, week 2)

Your name and student number must be on each page.

### Exercise 1

What is the ARC assembly instruction for the machine code (hexadecimal): 82806004

### Exercise 2

We are not the only institute that used the ARCTools. The following assignment is based on assignment 2 of [https://www.radford.edu/~aaray/ITEC\\_352/Homework\\_files/HW3.pdf](https://www.radford.edu/~aaray/ITEC_352/Homework_files/HW3.pdf)

A section of ARC assembly code is shown below. What does it do? Express your answer in terms of actions it goes through. Does it add up numbers or clear something out? Does it simulate a for loop, a while loop, or something else? Assume that a and b are memory locations that are defined elsewhere in the code.

**Test it in the ARCTools using correct values for a, b and c**

```
.begin
.org 0
Y:   ld [k], %r1
      addcc %r1,-4, %r1
      st %r1, [k]
      bneg X
      ld [%r1+a], %r2
      ld [b], %r3
      addcc %r2, %r3, %r4
      st %r4, [%r1+c]
      ba Y
X:   halt
k:   40
a:   ???
b:   ???
c:   ???
.end
```

**Exercise 3**

Given is an array arr1 that contains numbers (size of a number is 32 bits). The end of the array is indicated with the integer value 0. Write an assembly program that count the number of occurrences of the decimal number 2 in this array. The number of occurrences must be stored in %r10. **Add comment!**

**Simulate the program for different arrays with the ARCTools**

```

    .begin
    .org 0
    sethi arr1, %r1
    srl %r1,10,%r1    ! the address of arr1 in register %r1

< your program >

finished:  halt
    .org 200
arr1: 5,4,3,6,2,3,4,2,1,2,0 ! end of row is indicated with 0
    .end

```

**Exercise 4**

**Note: you do not have to finish this on Thursday. But discuss this assignment with the other students!**

The ARC processor is extended with the ABS instruction.

ABS %rx, %ry with %ry ← absolute value of %rx, representation is two's complement

The *condition codes* may change.

%rx and %ry are registers from the register file (%r0 until %r31; %ry ≠ %r0)

The instruction format is:

op=10, rd=%ry, op3=101010, rs1=is not used, bit13=0,

bit12 until bit5=00000000, rs2=%rx

a) What is the decimal value of the start address of the instruction ABS in the microstore?

b) Give an efficient microprogramme for instruction ABS.

Use symbolic names in the fields (e.g. in field A %r6 instead of 00110). Fields that are not used must be marked with '-' (don't care). The memory address space from 600-650 is not used by other instructions.

address	A	Amux	B	Bmux	C	Cmux	Rd	Wr	ALU	Cond	Jump addr